Guide de développement – Appel de l'API pour accès à la Prestation F375A / P375A de demande ponctuelle des données du compteur en infra-journalier « Infra-J »

Identification : **Enedis-GUI-CF_009E**
Version : **1**
Nb. de pages : **21**

Version	Date d'application	Nature de la modification	Annule et remplace
1	18/06/2019	Création du document	-

Document(s) associé(s) et annexe(s) :

- Enedis-FOR-CF_055E.xlsx : Formulaire de demande d'habilitation aux API Enedis
- Enedis-NOI-CF_107E : Procédure de déchiffrement des documents émis par Enedis sur les canaux numériques d'un Client ENTREPRISE

Résumé / Avertissement
Ce document à l'attention des Clients ENTREPRISES (de leurs services informatiques) a pour vocation d'aider au développement de l'appel de l'API Enedis permettant d'accéder au service de données « Infra-J » (Prestation F375A/P375A).

Pour rappel le service de données « Infra J » permet de disposer pour un point de connexion actif :

- Des données de courbes de charge et de tension
- Des données de gestion de la tarification dynamique
- Des données de mesure des flux d'énergie par poste de la période contractuelle courante

Enedis - Tour Enedis
34 place des Corolles
92079 Paris La Défense Cedex
enedis.fr

SA à directoire et à conseil de surveillance
Capital de 270 037 000 euros
R.C.S. de Nanterre 444 608 442
Enedis est certifié ISO 14001 pour l'environnement

SOMMAIRE

1. Les 2 plateformes API : Homologation et Production 3

2. Autorisation d'accès aux API Enedis 3

2.1. Introduction 3
2.2. Récupération d'un jeton d'accès 4
2.3. Utilisation d'un jeton d'accès 9
2.4. Renouvellement d'un jeton d'accès 10
2.5. Révocation d'un jeton d'accès 11

3. Utilisation de l'API « Demande d'accès Infra-Journalier aux données télé-relevables » 11

3.1. Demande d'enregistrement de l'application cliente auprès d'Enedis 11
3.2. Authentification 12
3.3. Appel de l'API 12
3.4. Réponse aux appels 13
3.5. Cas d'erreur et codes retour http 14
3.6. Swagger (fichier 20190521_daily_metering_data_request_v1.json) 16

1. Les 2 plateformes API : Homologation et Production

Il existe actuellement 2 plateformes d'API : une plateforme d'homologation exposant des API adossées à des données de test et une plateforme de production donnant accès aux API qui exposent les données de production.
Ces deux plateformes seront utilisées lors du processus d'accès à la production :

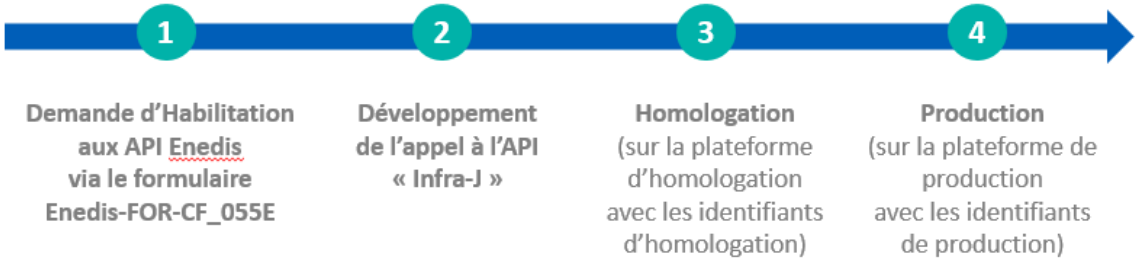


Figure 1 - Processus de mise à disposition de l'API

Dans la suite de ce document, les appels aux requêtes sont préfixés par le paramètre {endpoint_api}.
Selon la plateforme sur laquelle l'application cliente s'interface, il faudra remplacer ce paramètre par :

- {endpoint_api} : https://gw.hml.api.enedis.fr afin d'accéder aux API de la plateforme d'homologation
- {endpoint_api} : https://gw.prd.api.enedis.fr afin d'accéder aux API de la plateforme de production

2. Autorisation d'accès aux API Enedis

2.1. Introduction

Pour que votre application puisse accéder aux API d'Enedis, elle doit fournir la preuve qu'elle est bien autorisée à le faire.

Cette autorisation est matérialisée par un jeton d'accès, aussi nommé *acces token*, qui doit être présenté par l'application lors de chaque appel d'une API.

Cette rubrique explique quelles requêtes instancier pour récupérer un jeton d'accès aux API Enedis.

Enedis a choisi d'implémenter le protocole OAUTH 2.0, un standard du web, pour gérer l'accès à ses API.
Le principe de ce protocole est de fournir un jeton d'accès à un client (logiciel ou application qui effectue des demandes à un serveur) afin d'accéder à des informations protégées, appelées ressources.

Ces ressources sont matérialisées par des **API**.

Afin de protéger le jeton d'accès et les données, les échanges entre l'application et le serveur d'autorisation et de ressource doivent se faire en utilisant le protocole de sécurisation TLS (Transport Layer Security) qui permet de chiffrer des données échangées sur internet, c'est-à-dire en https.

Au préalable, l'application cliente s'est enregistrée auprès de ce serveur et a négocié avec lui l'accès à un ensemble d'API. Cet ensemble est appelé « scope ».

A l'issue de cet enregistrement, le développeur de l'application cliente a récupéré des identifiants d'accès (communiqués par Enedis via retour du formulaire **Enedis-FOR-CF_055E.xlsx**) que l'application devra présenter à chaque fois qu'elle veut obtenir un jeton d'accès.

Le jeton d'accès a une durée de vie limitée dans le temps définie par le serveur d'autorisation. Il peut être révoqué par le détenteur de la ressource ce qui annulera l'autorisation d'accès à ses ressources pour le client. Le jeton d'accès permet

d'accéder uniquement au scope d'API préalablement négocié lors de la souscription de l'application aux API, autrement dit, lors de l'enregistrement de l'application.

A chaque fois que l'application accédera à une API, elle présentera donc son jeton qui sera vérifié par le serveur d'autorisation. Si le jeton est valide, l'application pourra consulter les données, sinon elle ne sera pas autorisée et recevra en retour un code statut « 401 Unauthorized » ou « 403 Forbidden » selon le cas de figure.

Les causes d'invalidité du jeton peuvent être multiples mais la cause la plus courante est que sa date de validité est expirée. Dans ce cas, l'application devra demander un nouveau jeton en rejouant le protocole d'autorisation.

Parmi les scénarios d'autorisation d'accès offerts par le protocole OAUTH 2.0, Enedis a choisi d'implémenter celui de l'autorisation via un code (Authorization Code Grant Flow), qui nécessite l'authentification de l'utilisateur.

2.2. Récupération d'un jeton d'accès

Pour récupérer un jeton d'accès il est nécessaire de dérouler les étapes suivantes :

- Récupération d'un code d'autorisation
- Echange du code obtenu contre un jeton

Attention, lors de la récupération de mon jeton, je dois indiquer une URL de callback :

- Cette URL doit être **identique à celle que j'ai indiquée lors de la demande d'enregistrement de mon application** (dans le formulaire **Enedis-FOR-CF_055E.xlsx**)
- L'URL doit être protégée et accessible uniquement en **https**
- La longueur de l'URL de callback n'est pas limitée, mais **il est recommandé de ne pas dépasser 2000 caractères** pour la longueur totale d'appel à l'API, à cause de la limitation de certains navigateurs.

1ère étape - Récupération d'un code d'autorisation

Lancer dans le navigateur de votre choix la requête suivante :

```
GET
{endpoint_api}/v1/oauth2/authorize?client_id={client_id}&response_type={response_type}&redirect_uri={redirect_uri}&user_type={user_type}
```

Avec :

- {client_id} qui est l'identifiant de l'application fourni une fois votre application enregistrée. Ce champ est obligatoire.
- {response_type} qui doit avoir la valeur "code". Ce champ est obligatoire.
- {redirect_uri} qui est l'URL de "callback" fournie lors de votre demande d'habilitation aux API d'Enedis. Ce champ est obligatoire.
- {user_type} qui doit prendre la valeur "both".

Vous serez alors redirigé vers la page d'authentification de l'annuaire externe Enedis :

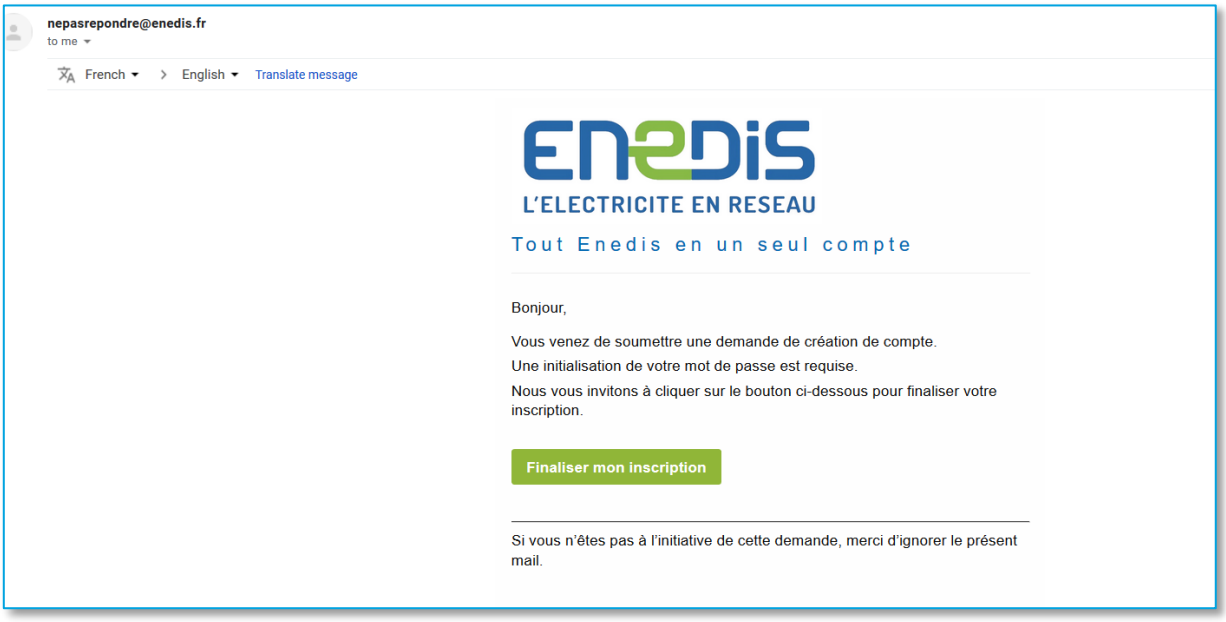
A cette étape, il y a deux cas à considérer :

1. **Authentification sur la plateforme de test**
2. **Authentification sur la plateforme de production**

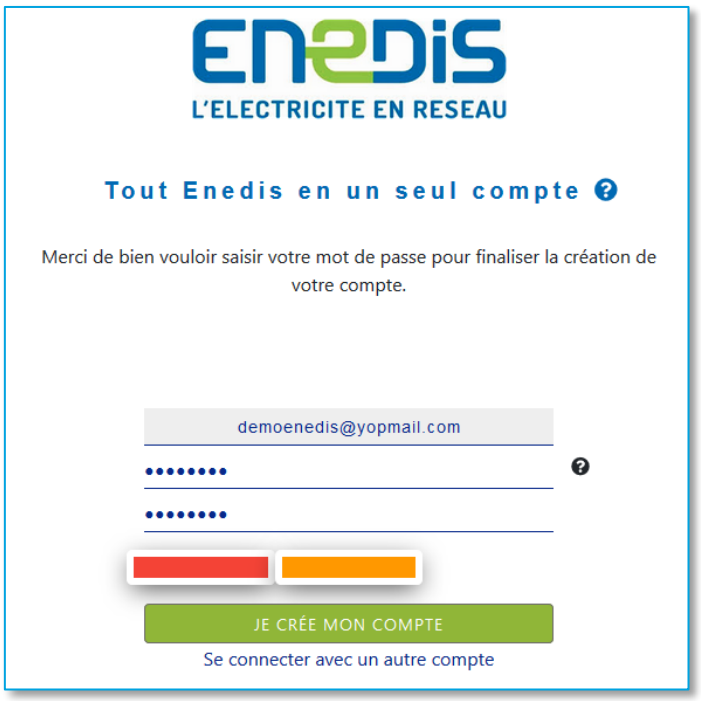
1. Pour l'authentification sur la plateforme de test, il vous sera nécessaire de créer un compte pour vous enregistrer dans l'annuaire externe Enedis de test :
   a. Renseignez l'email souhaité, s'il s'agit de votre première authentification, ou l'email du compte créé précédemment, puis validez le CAPTCHA et cliquez sur « Suivant »

b. S'il s'agit de votre première authentification, donc que votre email n'est pas encore connu de l'annuaire externe Enedis, renseignez un nom et prénom puis cliquez sur « Je crée mon compte »

c. Un email de confirmation vous sera alors envoyé, afin de finaliser la création de votre compte. Cliquez alors sur « Finaliser mon inscription »



d. Vous serez ensuite redirigé vers la page d'initialisation de votre mot de passe pour finaliser la création de votre compte. Pour terminer, cliquez sur « Je créé mon compte », puis sur « Accéder à mon application »



e. Saisissez enfin les identifiants de votre nouveau compte, ou de votre compte existant pour obtenir votre code

2. Pour l'authentification sur la plateforme de production, saisissez vos identifiants (email et mot de passe) de connexion à votre Espace Client Entreprise.

Si l'authentification est valide, le protocole d'autorisation vous redirigera vers votre URL de callback, suivie de votre code :

```
{redirect_uri}?code={code}
```

Avec :

- {redirect_uri} qui est l'URL de "callback" fourni lors de votre demande d'enregistrement de votre application (formulaire Enedis-FOR-CF_055E.xlsx)
- {code} qui est le code à échanger contre un jeton. Il a une durée de vie de quelques minutes.

2ème étape - Echange du code obtenu contre un jeton :

Pour l'échange du code d'autorisation contre un jeton d'accès, il faut initier la requête suivante :

```
POST {endpoint_api}/v1/oauth2/token?redirect_uri={redirect_uri}

Content-Type: application/x-www-form-urlencoded

client_id={client_id}&client_secret={client_secret}&grant_type=authorization_code&code={code}
```

Avec :

- {redirect_uri} qui est l'URL de "callback" fournie lors de votre demande d'habilitation aux API d'Enedis.

Transmission des variables en form-urlencoded :

- {client_id} qui est l'identifiant unique de l'application appelante. Ce champ est obligatoire.
- {client_secret} qui est le code unique de l'application appelante, associé au client_id. Ce champ est obligatoire.
- grant_type=authorization_code. Ce champ est obligatoire.
- {code} qui est le code obtenu lors de l'étape précédente. Ce champ est obligatoire.

Le retour est un JSON de la forme suivante :

```
{
    "access_token": {access_token},
    "token_type": "Bearer",
    "expires_in": {expires_in},
    "refresh_token": {refresh_token},
    "scope": {scope}
    "refresh_token_issued_at": {refresh_token_issued_at},
    "issued_at": {issued_at}
}
```

Avec :

- {access_token} qui est le jeton d'accès à transmettre pour la consommation des API exposées par la plateforme d'API.
- {token_type} qui donne le type de jeton généré. Dans notre cas, le type du jeton est « Bearer ».
- {expires_in} qui correspond à la durée du jeton avant qu'il n'expire (en secondes). Par défaut, le jeton dure 3 jours. {refresh_token} est le jeton de renouvellement à échanger contre un nouveau jeton d'accès «access_token » lorsque le jeton d'accès en cours arrive à échéance. Par défaut, ce jeton de renouvellement dure 6 jours.
- {scope} est le paramètre qui renvoie le périmètre d'action possible sur les différentes API de l'utilisateur authentifié.
- {refresh_token_issued_at} est le timestamp en millisecondes de la date d'émission du refresh_token.
- {issued_at} est le timestamp en millisecondes de la date d'émission du token.

Le retour est un code http **200** si tout s'est bien déroulé, un code **400** si la requête est mal formulée et un code **500** dans le cas d'une erreur côté serveur.

2.3. Utilisation d'un jeton d'accès

Une fois un jeton d'accès obtenu, il suffit de l'ajouter à chaque appel d'API dans le header Authorization, tel que dans l'exemple ci-dessous :

```
POST {endpoint_api}/v1/daily_metering_data_request?parametre1=xxx&parametre2=yyyy
Authorization: Bearer {access_token}
```

Avec :

- {access_token} qui est le jeton obtenu précédemment à l'échange du code. Ce header est obligatoire.

2.4. Renouvellement d'un jeton d'accès

Lorsque votre jeton d'accès en cours arrive à échéance, vous pouvez le renouveler sans avoir à demander un nouveau code au préalable, en utilisant le jeton de renouvellement fourni avec votre jeton d'accès.
Ceci est possible dans la limite de la durée de validité de votre jeton de renouvellement, qui pour rappel est de 6 jours. Dans le cas ou votre jeton de renouvellement serait aussi expiré, il vous faudra vous authentifier de nouveau Cf. 2.2 « Récupération d'un jeton d'accès » :

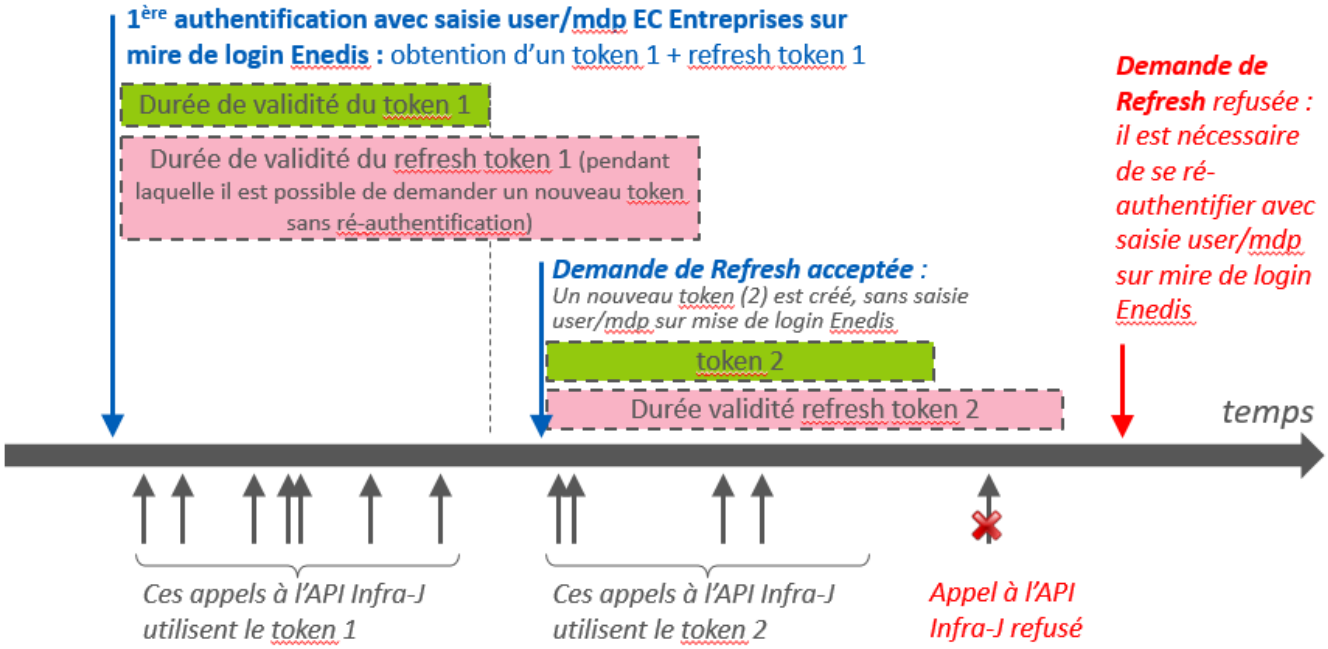


Figure 2 - Schéma récapitulatif du fonctionnement des jetons d'accès

Le renouvellement d'un jeton se fait tel que ci-dessous :

```
POST {endpoint_api}/v1/oauth2/token?redirect_uri={redirect_uri}
Content-Type: application/x-www-form-urlencoded

client_id={client_id}&client_secret={client_secret}&grant_type=refresh_token&refresh_token={refresh_token}
```

Avec :

- {redirect_uri} qui est l'URL de "callback" fournie lors de votre demande d'habilitation aux API d'Enedis. Ce champ est obligatoire.

Transmission des variables en form-urlencoded :

- {client_id} qui est l'identifiant unique de l'application appelante. Ce champ est obligatoire.
- {client_secret} qui est le code unique de l'application appelante, associé au client_id. Ce champ est obligatoire.
- grant_type=refresh_token. Ce champ est obligatoire.

- {refresh_token} qui est le jeton de renouvellement obtenu lors de l'étape d'autorisation par code. Ce champ est obligatoire.

Le retour est un JSON de la forme suivante :

```
{
    "access_token": {access_token},
    "token_type": "Bearer",
    "expires_in": {expires_in},
    "refresh_token": {refresh_token},
    "scope": {scope}
    "refresh_token_issued_at": {refresh_token_issued_at},
    "issued_at": {issued_at}
}
```

Avec :

- {access_token} qui est le jeton d'accès à transmettre pour la consommation des API exposées par la plateforme d'API
- {token_type} qui donne le type de jeton généré. Dans notre cas, le type du jeton est « Bearer ».
- {expires_in} qui est la durée du jeton avant qu'il n'expire (en secondes). Par défaut, le jeton dure 3h30
- {refresh_token} qui est le jeton à échanger contre un nouveau jeton d'accès «access_token » lorsque le jeton d'accès en cours arrive à échéance
- {scope} est le paramètre qui renvoie le périmètre d'action possible sur les différentes API de l'utilisateur authentifié.
- {refresh_token_issued_at} est le timestamp en millisecondes de la date d'émission du refresh_token.
- {issued_at} est le timestamp en millisecondes de la date d'émission du token.

Le retour est un code http **200** si tout s'est bien déroulé, un code **400** si la requête est mal formulée et un code **500** dans le cas d'une erreur côté serveur.

2.5. Révocation d'un jeton d'accès

Dans certains cas, il peut être nécessaire de révoquer (supprimer) un jeton d'accès, via la requête ci-dessous :

```
POST {endpoint_api}/v1/oauth2/revoke
Content-Type: application/x-www-form-urlencoded

token={token}&token_type_hint=access_token
```

Transmission des variables en form-urlencoded :

- {token} : token à supprimer.
- token_type_hint=access_token : type de token à supprimer.

Le retour est un code http 200 si tout s'est bien déroulé, un code 400 si la requête est mal formulée et un code 500 dans le cas d'une erreur côté serveur.

3. Utilisation de l'API « Demande d'accès Infra-Journalier aux données télé-relevables »

3.1. Demande d'enregistrement de l'application cliente auprès d'Enedis

Pour demander l'enregistrement de votre application auprès d'Enedis afin qu'elle puisse utiliser la Prestation de demande ponctuelle d'accès aux données d'un compteur en infra-journalier, vous devez :

1. Télécharger le formulaire « Formulaire de demande d'habilitation aux API d'Enedis » référence **Enedis-FOR-CF_055E** disponible sur le site www.enedis.fr
2. Renseigner ce formulaire en indiquant renseignant toutes les cellules sur fond vert (les coordonnées de l'interlocuteur habilité à recevoir les éléments d'enregistrement, et, pour chacune des Entreprises (au sens SIREN) associées aux compteurs que vous souhaitez interroger :
    a. Le n° de SIREN de l'Entreprise, et son nom (Raison Sociale)

b. L'URL de callback définie au chapitre 2.2

3. Sauvegarder le formulaire sous le nom proposé
4. Envoyer le formulaire sous forme de pièce jointe par mail à l'adresse « dataconsoelec@enedis.fr », en précisant l'objet suivant : [SOUSCRIPTION_INFRAJ] suivi de votre raison sociale (par exemple : « [SOUSCRIPTION_INFRAJ] Mon entreprise »)

Vous recevrez par mail (à l'adresse email de l'interlocuteur) le formulaire complété avec pour chaque SIREN les identifiants « client_id » et « client_secret ».

Attention : afin de protéger ces informations sensibles, le formulaire que vous recevrez sera chiffré avec 7-Zip et la clé de déchiffrement sera envoyée par SMS au numéro de mobile indiqué dans le formulaire de demande d'habilitation. Pour le déchiffrer, utiliser 7-Zip ou Winzip et saisir le mot de passe.

3.2. Authentification

Avant d'utiliser l'API, il est nécessaire de s'authentifier afin d'obtenir un jeton d'accès. Cf. 2.2 « Récupération d'un jeton d'accès »

3.3. Appel de l'API

Un appel à cette API permet au demandeur de disposer, pour un point de connexion actif, des données brutes de courbes (de charge, de tension), d'index, et/ou de paramètres de tarifs dynamiques issues du compteur, avec une fraicheur des données inférieure à 30 min.

Pour effectuer une demande, initiez une requête telle que ci-dessous :

```
POST {endpoint_api}/v1/daily_metering_data_request

Authorization: Bearer {access_token}

{
  "usage_point_Id": {usage_point_Id},
  "holder_agreement_declaration": [
    {
      "holder_name": {holder_agreement_declaration},
      "production_agreement": {production_agreement},
      "consumption_agreement": {consumption_agreement}
    }
  ],
  "contact_canal_id": {contact_canal_id},
  "production": {production},
  "consumption": {consumption},
  "load_curve": {load_curve},
  "index": {index},
  "dynamic_tarification_parameters": {dynamic_tarification_parameters}
}
```

Avec :

- {usage_point_Id} : Identifiant PRM du point faisant l'objet de la demande. Ce champ est obligatoire.
- {holder_agreement_declaration} : Tableau de déclaration de détenir l'accord du (ou des) titulaire(s) du point pour accéder à ses (leurs) données d'injection et/ou de soutirage. La présence de ce tableau est obligatoire. A renseigner systématiquement, même si votre entreprise est elle-même titulaire du point. NB : Dans le cas où vous seriez titulaire en soutirage du point mais pas en injection (ou inversement), poussez une deuxième entrée dans le tableau, en précisant le nom de l'autre entreprise titulaire.
    - {holder_name} : Dénomination sociale du titulaire en injection et/ou soutirage du point.
    - {production_agreement} : Booléen précisant s'il s'agit d'une déclaration d'accord sur les données en injection ("true" si la déclaration d'accord porte sur les données en injection, "false" sinon).

  - {consumption_agreement} : Booléen précisant s'il s'agit d'une déclaration d'accord sur les données en soutirage ("true" si la déclaration d'accord porte sur les données en soutirage, "false" sinon).
- {contact_canal_id} : Canal de mise à disposition des fichiers de données demandées ; cet identifiant de canal est disponible sur votre espace entreprise, sur la page d'affichage des canaux définis « Mes canaux de contact »
- {production} : Booléen précisant si les données en injection sont demandées ("true" pour demander les données en injection, "false" sinon). Ceci s'applique aux données de courbe de charge/tension et d'index.
- {consumption} : Booléen précisant si les données en soutirage sont demandées ("true" pour demander les données en soutirage, "false" sinon). Ceci s'applique aux données de courbe de charge/tension, d'index et aux paramètres de tarification dynamique.
- {load_curve} : Booléen précisant si les données des courbes de charge/tension sont demandées ("true" pour demander les données des courbes de charge/tension, "false" sinon).
- {index} : Booléen précisant si les données d'index sont demandées ("true" pour demander les données d'index, "false" sinon)
- {dynamic_tarification_parameters} : Booléen précisant si les données de gestion de la tarification dynamique sont demandées ("true" pour demander les paramètres de tarification dynamique, "false" sinon)

Important : L'ensemble des champs du body sont obligatoires et à renseigner au format string.

Exemple de requête :

```
POST {endpoint_api}/v1/daily_metering_data_request

Authorization: Bearer F7bRKKkcOukI0C799Ak4lVp7z9uAkTj6Of4SIkbt62JMx5LAaiMHZQ

{
  "usage_point_Id": "11111111111111",
  "holder_agreement_declaration": [
    {
      "holder_name": "NOM",
      "production_agreement": "false",
      "consumption_agreement": "true"
    }
  ],
  "contact_canal_id": "11111111",
  "production": "false",
  "consumption": "true",
  "load_curve": "true",
  "index": "true",
  "dynamic_tarification_parameters": "false"
}
```

3.4. Réponse aux appels

En réponse aux appels, l'application reçoit un code retour HTTP (code de statut complété d'un contenu).
Dans le cas d'une réponse réussie, le code retour est un **200** OK complété par un body de la forme suivante :

```
{
  "request_Id": {request_Id}
}
```

Avec :

- {request_Id} : Identifiant de la demande. Cet identifiant sera présent dans le nom du fichier généré à l'issue du traitement.

Exemple :

```
{
  "request_Id": "M005MSEJ"
}
```

3.5. Cas d'erreur et codes retour http

En réponse aux appels, l'application reçoit un code retour HTTP (code de statut complété d'un message).
Dans le cas d'une réponse en erreur, les codes retour sont de type **4xx** (pour les erreurs client) ou **5xx** (pour les erreurs techniques ou serveur), complétés par un body de la forme suivante :

```
{
    "error" : "description_courte",
    "error_description" : "description longue"
}
```

Les valeurs des paramètres "*error*" et "*error_description*" dépendent du cas d'erreur rencontré.
Ci-dessous la liste des cas d'erreurs possibles et codes et messages associés :

Cas d'erreur	Code	Error	Description
Champ obligatoire manquant ou non renseigné dans le body de la requête.	**400**	Invalid_request	Empty header, bad request format, partial or incorrect data
Champ mal valorisé, ne respectant pas les valeurs de l'énum. Ex : consumption_agreement ni à « true » ni à « false ».	**400**	Invalid_request	Empty header, bad request format, partial or incorrect data
Les données d'injection sont demandées alors que vous ne déclarez pas détenir l'accord du titulaire d'accéder à ces données.	**400**	ADAM-ERR0117	Votre demande nécessite l'autorisation expresse du client pour l'injection.
Les données de soutirage sont demandées alors que vous ne déclarez pas détenir l'accord du titulaire d'accéder à ces données.	**400**	ADAM-ERR0118	Votre demande nécessite l'autorisation expresse du client pour le soutirage.
Une demande strictement identique a été faite il y a moins de 30 minutes	**400**	ADAM-ERR0110	La demande ne peut pas aboutir, vous avez une demande strictement identique en cours qui date de moins de 30 minutes.
Le point qui fait l'objet de la demande n'est pas encore en service.	**400**	ADAM-ERR0007	La demande ne peut pas aboutir car le point n'est pas en service.

Le dispositif de comptage du point qui fait l'objet de la demande porte une particularité différente de standard.	**400**	ADAM-ERR0010	La demande ne peut pas aboutir, le dispositif de comptage du point porte une particularité différente de standard.
Une prestation de raccordement provisoire courte durée est en cours sur le point qui fait l'objet de la demande.	**400**	ADAM-ERR0011	La demande ne peut pas aboutir car une prestation de raccordement provisoire courte durée est en cours sur le point.
Le point qui fait l'objet de la demande est coupé.	**400**	ADAM-ERR0012	La demande ne peut pas aboutir car le point est coupé.
Le point qui fait l'objet de la demande est sur un contrat producteur.	**400**	ADAM-ERR0013	La demande ne peut pas aboutir car le point est sur un contrat producteur.
Le point qui fait l'objet de la demande est un point fictif de regroupement TURPE.	**400**	ADAM-ERR0019	La demande ne peut pas aboutir car le point est un point fictif de regroupement TURPE.
Le point qui fait l'objet de la demande ne dispose pas d'un compteur téléopérable.	**400**	ADAM-ERR0023	La demande ne peut pas aboutir car le point ne dispose pas d'un compteur téléopérable. Une intervention comptage doit être demandée (changement de compteur ou intervention sur le dispositif de télécommunication).
L'installation de comptage du point qui fait l'objet de la demande est bloquée.	**400**	ADAM-ERR0024	La demande ne peut pas aboutir car l'installation de comptage du point est bloquée.
Le point qui fait l'objet de la demande n'est pas alimenté.	**400**	ADAM-ERR0059	La demande ne peut pas aboutir, le point n'est pas alimenté.
Le point qui fait l'objet de la demande est limité.	**400**	ADAM-ERR0094	La demande ne peut pas aboutir car le point est Limité.
Le média de relevé qui fait l'objet de la demande n'est pas un boitier IP.	**400**	ADAM-ERR0106	La demande ne peut pas aboutir car le média de relevé n'est pas un boitier IP.
Le SI contractuel du point qui fait l'objet de la demande n'est pas compatible avec la demande.	**400**	ADAM-ERR0111	La demande ne peut pas aboutir, le SI contractuel du point n'est pas compatible avec la demande.

La donnée n'est pas disponible sur le type de compteur qui fait l'objet de la demande.	**400**	ADAM-ERR0112	La demande ne peut pas aboutir, la donnée n'est pas disponible sur ce type de compteur.
Le compteur qui fait l'objet de la demande n'est pas de type PME-PMI, ICE, ICE4Q ou SAPHIR.	**400**	ADAM-ERR0115	Le compteur n'est pas de type PME-PMI, ICE, ICE4Q ou SAPHIR.
Le titulaire du point qui fait l'objet de la demande ne peut pas être identifié.	**400**	ADAM-ERR0067	La demande ne peut pas aboutir car le titulaire du point ne peut pas être identifié.
L'entreprise demandeuse n'est pas reconnue comme titulaire du point à la date de la demande.	**400**	ADAM-ERR0109	Aucun rôle n'est actif à date de traitement.
L'entreprise demandeuse n'est pas reconnue comme titulaire du point à la date de la demande.	**400**	ADAM-ERR0120	La demande ne peut pas aboutir, l'entreprise demandeuse n'est pas reconnue comme référent du point à date de la demande.
Mauvais jeton ou jeton arrivé a expiration	**403**	unauthorized_user	User is not authorized to access this resource
Quota d'appels dépassé	**429**	sla_exceeded	Number of authorized requests exceeded
Erreur survenue lors du traitement	**500**	technical_error	Technical error. Please try later.
Service temporairement indisponible	**503**	service_unavailable	Service unavailable for the moment. Please try later

3.6. Swagger (fichier 20190521_daily_metering_data_request_v1.json)


```
{
  "swagger" : "2.0",
  "info" : {
    "description" : "Permet au demandeur de disposer, pour un point de connexion actif, des données brutes de courbe de charge, courbe de tension et paramètres de tarifs dynamiques issues du compteur, avec une fraicheur des données de 30 min.",
    "version" : "v1",
    "title" : "Demande d'accès Infra-Journalier aux données télé-relevables",
    "contact" : { }
  },
  "host" : "gw.hml.api.enedis.fr",
  "schemes" : [ "https" ],
  "consumes" : [ "application/json" ],
```

```
  "produces" : [ "application/json" ],
  "paths" : {
    "/v1/daily_metering_data_request" : {
      "post" : {
        "summary" : "Demande de transmission de données de mesures infra-journalières",
        "consumes" : [ ],
        "parameters" : [ {
          "name" : "Authorization",
          "in" : "header",
          "required" : true,
          "type" : "string",
          "description" : "Token Oauth délivré à l'application cliente<br/>\nFormat Bearer Token"
        }, {
          "name" : "Accept",
          "in" : "header",
          "required" : false,
          "type" : "string",
          "description" : "Format de réponse attendu par l'application",
          "x-example" : "application/json"
        }, {
          "name" : "Content-Type",
          "in" : "header",
          "required" : false,
          "type" : "string",
          "description" : "Fomat du body de la requête",
          "x-example" : "application/json"
        }, {
          "name" : "body",
          "in" : "body",
          "required" : true,
          "schema" : {
            "type" : "object",
            "required" : [ "consumption", "contact_canal_id", "dynamic_tarification_parameters", "holder_agreement_declaration", "index", "load_curve", "production", "usage_point_Id" ],
            "properties" : {
              "usage_point_Id" : {
                "type" : "string",
                "description" : "Identifiant PRM du point faisant l'objet de la demande"
              },
              "holder_agreement_declaration" : {
                "type" : "array",
                "description" : "Déclaration de détenir l'accord du (ou des) titulaire(s) du point pour accéder à ses (leurs) données d'injection et/ou de soutirage",
                "items" : {
                  "required" : [ "consumption_agreement", "holder_name", "production_agreement" ],
                  "type" : "object",
                  "properties" : {
                    "holder_name" : {
                      "type" : "string",
                      "description" : "Dénomination sociale du titulaire en injection et/ou soutirage du point"
                    },
                    "production_agreement" : {
```

```
                    "type" : "string",
                    "description"   :   "Booléen   précisant   s'il   s'agit   d'une
déclaration d'accord sur les données en injection : <br/>\n- « true » si la
déclaration d'accord porte sur les données en injection \n- « false » sinon",
                    "enum" : [ "true", "false" ],
                    "example" : "true"
                  },
                  "consumption_agreement" : {
                    "type" : "string",
                    "description"   :   "Booléen   précisant   s'il   s'agit   d'une
déclaration d'accord sur les données en soutirage : <br/>\n- « true » si la
déclaration d'accord porte sur les données en soutirage\n- « false » sinon",
                    "enum" : [ "true", "false" ],
                    "example" : "true"
                  }
                }
              }
            },
            "contact_canal_id" : {
              "type" : "string",
              "description" : "Canal de mise à disposition des fichiers de données
demandées"
            },
            "production" : {
              "type" : "string",
              "description" : "Booléen précisant si les données en injection sont
demandées :<br/>\n- «true» pour demander les données en injection<br/>\n- «false»
sinon<br/>\n\nCeci s'applique aux données de courbe de charge, de tension et
d'index.",
              "enum" : [ "true", "false" ],
              "example" : "true"
            },
            "consumption" : {
              "type" : "string",
              "description" : "Booléen précisant si les données en soutirage sont
demandées :<br/>\n- «true» pour demander les données en soutirage<br/>\n- «false»
sinon<br/>\n\nCeci s'applique aux données de courbe de charge et de tension, d'index
et aux paramètres de tarification dynamique.",
              "enum" : [ "true", "false" ],
              "example" : "true"
            },
            "load_curve" : {
              "type" : "string",
              "description" : "Booléen précisant si les données des courbes de
charge et de la courbe de tension sont demandées :<br/>\n- «true» pour demander les
données des courbes de charge et la courbe de tension <br/>\n- «false» sinon",
              "enum" : [ "true", "false" ],
              "example" : "true"
            },
            "index" : {
              "type" : "string",
              "description"   :   "Booléen   précisant   si   les   données   d'index   sont
demandées :<br/>\n- «true» pour demander les données d'index <br/>\n- «false» sinon",
              "enum" : [ "true", "false" ],
              "example" : "true"
            },
            "dynamic_tarification_parameters" : {
              "type" : "string",
```

```
                "description" : "Booléen précisant si les données de gestion de la tarification dynamique sont demandées :\n- «true» pour demander les paramètres de tarification dynamique <br/>\n- «false» sinon <br/>",
                "enum" : [ "true", "false" ],
                "example" : "true"
              }
            }
          }
        } ],
        "responses" : {
          "200" : {
            "description" : "Status 200",
            "schema" : {
              "required" : [ "request_Id" ],
              "type" : "object",
              "properties" : {
                "request_Id" : {
                  "type" : "string",
                  "description" : "Identifiant de la demande. Cet identifiant sera dans le nom du fichier généré à l'issu du traitement."
                }
              }
            }
          },
          "400" : {
            "description" : "Status 400",
            "schema" : {
              "type" : "object",
              "properties" : {
                "error" : {
                  "type" : "string",
                  "description" : "Code de l'erreur retourné par le système.",
                  "enum" : [ "Invalid_request", "ADAM-ERR0117", "ADAM-ERR0118", "ADAM-ERR0116", "ADAM-ERR0110", "ADAM-ERR0007", "ADAM-ERR0010", "ADAM-ERR0011", "ADAM-ERR0012", "ADAM-ERR0013", "ADAM-ERR0019", "ADAM-ERR0023", "ADAM-ERR0024", "ADAM-ERR0059", "ADAM-ERR0094", "ADAM-ERR0106", "ADAM-ERR0111", "ADAM-ERR0112", "ADAM-ERR0115", "ADAM-ERR0067", "ADAM-ERR0109", "ADAM-ERR0120" ],
                  "example" : "Invalid_request"
                },
                "error_description" : {
                  "type" : "string",
                  "description" : "Description de l'erreur correspondant au code.",
                  "enum" : [ "Empty header, bad request format, partial or incorrect data", "Votre demande nécessite l'autorisation expresse du client pour l'injection.", "Votre demande nécessite l'autorisation expresse du client pour le soutirage.", "Le sens de la mesure (injection et/ou soutirage) doit être précisé dans la demande.", "La demande ne peut pas aboutir, vous avez une demande strictement identique en cours qui date de moins de 30 minutes.", "La demande ne peut pas aboutir car le point n'est pas en service.", "La demande ne peut pas aboutir, le dispositif de comptage du point porte une particularité différente de standard.", "La demande ne peut pas aboutir car une prestation de raccordement provisoire courte durée est en cours sur le point.", "La demande ne peut pas aboutir car le point est coupé.", "La demande ne peut pas aboutir car le point est sur un contrat producteur.", "La demande ne peut pas aboutir car le point est un point fictif de regroupement TURPE.", "La demande ne peut pas aboutir car le point ne dispose pas d'un compteur téléopérable. Une intervention comptage doit être demandée (changement de compteur ou intervention sur le dispositif de télécommunication).", "La demande ne peut pas aboutir car l'installation de comptage du point est bloquée.", "La demande ne peut pas aboutir, le point n'est pas alimenté.", "La demande ne peut pas aboutir car le point est Limité.", "La demande ne
```

```
peut pas aboutir car le média de relevé n'est pas un boitier IP.", "La demande ne peut pas aboutir, le SI contractuel du point n'est pas compatible avec la demande.", "La demande ne peut pas aboutir, la donnée n'est pas disponible sur ce type de compteur.", "Le compteur n'est pas de type PME-PMI, ICE, ICE4Q ou SAPHIR.", "La demande ne peut pas aboutir car le titulaire du point ne peut pas être identifié", "Aucun rôle n'est actif à date de traitement.", "La demande ne peut pas aboutir, l'entreprise demandeuse n'est pas reconnue comme référent du point à date de la demande." ],
                    "example" : "Empty header, bad request format, partial or incorrect data"
                }
              }
            }
          },
          "403" : {
            "description" : "Status 403",
            "schema" : {
              "type" : "object",
              "properties" : {
                "error" : {
                  "type" : "string",
                  "description" : "Code de l'erreur retourné par le système.",
                  "enum" : [ "unauthorized_user" ],
                  "example" : "unauthorized_user"
                },
                "error_description" : {
                  "type" : "string",
                  "description" : "Description de l'erreur correspondant au code.",
                  "example" : "User is not authorized to access this resource"
                }
              }
            }
          },
          "429" : {
            "description" : "L'application a dépassé son quota d'appels",
            "schema" : {
              "type" : "object",
              "properties" : {
                "error" : {
                  "type" : "string",
                  "description" : "Code de l'erreur retourné par le système.",
                  "example" : "sla_exceeded"
                },
                "error_description" : {
                  "type" : "string",
                  "description" : "Description de l'erreur correspondant au code.",
                  "example" : "Number of authorized requests exceeded"
                }
              }
            }
          },
          "500" : {
            "description" : "Status 500",
            "schema" : {
              "type" : "object",
              "properties" : {
                "error" : {
                  "type" : "string",
                  "description" : "Code de l'erreur retourné par le système.",
```

```
            "example" : "technical_error"
          },
          "error_description" : {
            "type" : "string",
            "description" : "Description de l'erreur correspondant au code.",
            "example" : "Technical error. Please try later."
          }
        }
      }
    },
    "503" : {
      "description" : "Le serveur est momentanément inaccessible",
      "schema" : {
        "type" : "object",
        "properties" : {
          "error" : {
            "type" : "string",
            "description" : "Code de l'erreur retourné par le système",
            "example" : "service_unavailable"
          },
          "error_description" : {
            "type" : "string",
            "description" : "Description de l'erreur correspondant au code.",
            "example" : "Service unavailable for the moment. Please try later"
          }
        }
      }
    }
  }
}
}
```